NUI Galway
OÉ Gaillimh

Semester 2 Examinations 2017/2018

Exam Code(s)	4BCT, 4ECE
Exam(s)	B.Sc. in Comp Sc. & Information Technology B.E. in Elect.& Comp. Engineering
Module Code(s)	CT420
Module(s)	Real-Time Systems
Paper No.	1
Repeat Paper	N
External Examiner(s)	Dr. Jacob Howe
Internal Examiner(s)	Prof. Michael Madden *Mr. Mohannad Alahmadi *Dr. Michael Schukat

Instructions: Answer Q1 and 1 other Q from section A.
Answer 2 questions from section B.
All questions carry equal marks.

Duration	2 hours
No. of Pages	6
Discipline	Information Technology

Requirements:
MCQ
Handout
Statistical/ Log Tables
Cambridge Tables
Graph Paper
Log Graph Paper
Other Materials

Release to Library: Yes [X] No [ ]

Section A

Q1

(i) Mouth to Ear delay is a key metric for Voice over IP. Explain what it is, why it is a key metric, and outline the main contributory elements on the sender, network and receiver.

[20]

(ii) Distinguish between Media Independent and Media Dependent Forward Error Correction (FEC) strategies for Voice over IP, using examples to illustrate your answer. Comment also on their respective impact on delay, bandwidth utilisation, and quality.

[10]

(iii) DASH is widely used for streaming multimedia. Explain briefly what it is, how it works and why it works well in today's best effort Internet.

[10]

Q2

(i) Many RTS have specific requirements for both time and timing synchronisation. Distinguish clearly between these two concepts.

[10]

(ii) Outline similarities and differences between the NTP and the PTP protocol. Your answer should also refer to Round Trip Delay (RTD) and Offset, and explain the meaning and calculation of both parameters.

[15]

(iii) The data below shows the output of the ntpq utility from 2 NTP clients, A and B. Based on the output, which client is most likely to deliver better Synchronisation? Your answer should comment on redundancy, offsets, delay, stratum level and reachability.

ClientA	**RefID**	**st**	**t**	**When**	**Poll**	**Reach**	**Delay**	**Offset**	**Jitter**
+server1	serverx	2	u	51	64	156	391.281	6.24	7.79
-server2	servery	2	u	49	64	372	353.217	10.435	1.663
*server3	serverz	2	u	50	64	356	85.688	2.465	2.666
+server4	.PPS.	1	u	49	64	357	66.369	1.858	2.105
ClientB	**RefID**	**st**	**t**	**When**	**Poll**	**Reach**	**Delay**	**Offset**	**Jitter**
+server5	serverm	2	u	45	64	377	36.345	2.24	3.458
-server6	servern	2	u	42	64	377	313.217	3.435	1.11
*server7	.PPS.	1	u	13	64	377	10.688	1.485	1.455
+server8	.PPS.	1	u	22	64	377	28.456	1.818	1.345
+server9	.GPS.	1	u	18	64	356	15.345	-2.654	2.34

[15]

Q3

(i) In the context of voice communication, distinguish between intrinsic and perceived Quality-of-Service (QoS), outlining also how they can be measured.

[10]

(ii) Distinguish between narrowband (NB), wideband (WB), and super-wideband (SWB) voice codecs, outlining the advantages/disadvantages of each.

[10]

(iii) Briefly describe the use of RTP and RTCP protocols? Differentiate between the sender report and receiver report in RTCP.

[20]

PTO

Section B

Q4

(i) You are asked to design the flight control computer for an interstellar space probe. This computer system is supposed to be ultra-redundant by incorporating different means of redundancy, i.e.

- mass storage redundancy via RAID-5,
- data i.e. RAM redundancy via a Single-Error Correcting Code (SEC), and
- appropriate measures to provide further hardware redundancy.

Outline in detail what such a system could look like, i.e. how all these technologies could be combined. In your answer explain in detail the operation of RAID-5, SEC and your suggested means of hardware redundancy.

[15]

Suggest the different types of errors that could be captured / compensated by your design, and using examples explain how the above three means of redundancy would make your system more robust.

[10]

(ii) Memory locking is an important feature of POSIX 4. Explain the code snippet below and use the 7-state process model in conjunction with an example to outline why memory locking is important for (hard) real-time systems.
Further on, discuss the consequences of memory locking if being applied too generously.

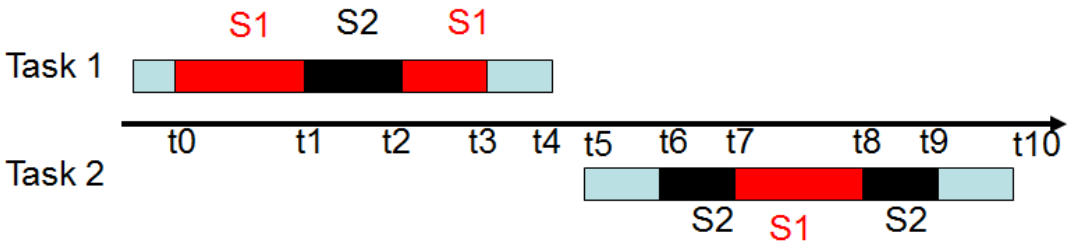
```
/*Main routine */
int main(void ){
/* Lock all process down */

mlockall(MCL_CURRENT|MCL_FUTURE);

… process code

munlockall();
return 0;
}
```

[15]

PTO

Q5

(i) The diagram below shows the timeline of two tasks, i.e. Task 1 and Task 2, which require at different times two semaphores, i.e. S1 and S2, in order to gain access to some critical resources. Note that Task 1 has a higher priority than Task 2.



Modify the timeline to enforce a deadlock, and explain the different steps that lead to this deadlock.

[5]

Using your example in (i) discuss in detail, how the *Priority Ceiling Protocol* can avoid a deadlock situation. Your answer should be complemented by a diagram of the resulting timeline.

[20]

(ii) Outline a potential *run-time problem* of the function f() below:

```
int f(int i)
{
        int n;
        if (i > 0) {
                n = -1;
        }
        return n;
}
```

[5]

How can this problem be fixed?

[2]

Use the function f() above as an example to explain the term *fault latency*.

[8]

PTO

Q6

(i) Using an example schedule consisting of 3 tasks, distinguish between the *RM scheduling algorithm* and the *EDF scheduling algorithm*.

[5]

Construct a task schedule, consisting of 4 tasks, that is EDF schedulable, but NOT RM schedulable.
Proof your design by calculating

- the Overall Utilisation U
- the task timeline
- an RM Schedulability Analysis

[20]

(ii) Summarise the main features of POSIX.4 signals.
Further on, outline the purpose and functionality of the function:
int sigqueue(pid_t pid, int sig, const union sigval value);

[15]